

**AMENDMENTS TO THE CLAIMS**

1-18. (Canceled)

19. (Currently Amended) A ~~data transfer~~ system for transferring data from a data source to multiple data sink objects in a computer system, wherein the data source is coupled to the computer, the system comprising:

a data source holding acquired data;

an interface for communicating with the data source to receive the data from the data source;

a computing device comprising:

a processor controlling a data processor~~for encapsulating that encapsulates~~ the data into a data object in ~~the~~ a first memory;

a secondary memory comprising a data server for:

transferring to the multiple data sink objects identification information identifying the data object, the data server providing a pointer indicating ~~the~~ a location of the data object in the memory to identify the data object, the multiple data sink objects accessing the data object using the identification information; and sharing the data object among the multiple data sink objects to prevent extraneous copies of the data.

20. (Original) The system of claim 19 wherein the data server includes a list listing the multiple data sink objects that are registered with the data server.

21. (Previously Presented) The system of claim 19 wherein the computer system provides a technical computing environment.

22. (Canceled)

23. (Previously Presented) The system of claim 19 further comprising at least one data listener object that is registered to a respective one of the multiple data sink objects.

24. (Currently Amended) The system of claim 23, wherein the respective one of the multiple data sink objects deletes each of the at least one data listener objects registered with the respective one of the multiple data sink objects when the respective one of the multiple data sink objects is deleted.

25. (Currently Amended) The system of claim 23, wherein the respective one of the multiple data sink objects notifies each of the at least one data listener objects registered with the respective one of the multiple data sink objects when the respective one of the multiple data sink objects is deleted.

26. (Currently Amended) The system of claim 23, wherein the respective one of the multiple data sink objects notifies each of the at least one data listener objects when the respective one of the multiple data sink objects is updated with a new data object.

27. (Original) The system of claim 19 wherein the data source provides data sequence continuously for a period of time.

28. (Original) The system of claim 19 wherein the data source provides a package of data, the package of data being used independently of other packages of data.

29. (Original) The system of claim 28 wherein the package of data includes a frame of image data.

30. (Original) The system of claim 28 wherein the package of data includes a scan of radar, sensor, or audio data, as well as network data packets.

31. (Previously Presented) The system of claim 19 wherein the data processor configures a maximum amount of memory that all data objects use[s] at a given instance of time.

32. (Currently Amended) The system of claim 19 ~~further comprising a processor for controlling the interface, the data processor, and the data server~~, wherein the processor is 64 bits or more.

33. (Original) The system of claim 19 wherein the interface, the data processor, and the data server are implemented independently of MATLAB.

34. (Currently Amended) A physical computer readable medium holding instructions executable in a computer system, wherein the computer system receives data from a data source and transfers the data to data sink objects, the medium holding:

one or more instructions for communicating with the data source to receive the data from the data source;

one or more instructions for encapsulating the data into a data object in ~~the a~~ memory;

one or more instructions for a data server object transferring to the data sink objects identification information identifying the data object, the data server object providing a pointer indicating ~~the a~~ location of the data object in the memory to identify the data object, the data sink objects accessing the data object using the identification information; and

one or more instructions for sharing the data object among the ~~multiple~~ data sink objects to prevent extraneous copies of the data.

35. (Original) The medium of claim 34 further comprising a data sink listener object that is registered with one or more of the data sink objects.

36. (Previously Presented) The medium of claim 34 wherein the computer system provides a technical computing environment.

37. (Previously Presented) The medium of claim 35 wherein the data sink listener object performs a task relating to a function of a respective one of the data sink objects.

38. (Previously Presented) The medium of claim 35 wherein the data sink listener object performs a task relating to a function of a respective one of the data sink objects on a thread of the data server object.

39. (Previously Presented) The medium of claim 35 wherein the data sink listener object performs a task relating to a function of a respective one of the data sink objects on a thread different from that of the data server object.

40. (Previously Presented) The medium of claim 34 wherein at least one of the data sink objects performs a function on a thread of the data server object.

41. (Previously Presented) The medium of claim 34 wherein at least one of the data sink objects performs a function on a thread different from that of the data server object.

42. (Original) The medium of claim 34 wherein the instructions are run independently of MATLAB.

43. (Original) The medium of claim 34 wherein the instructions are originated from code written with C programming language.

44. (Currently Amended) The medium of claim 34 wherein the instructions are originated from code written with an object-oriented programming language, the object-oriented programming language comprising one or more of such as C++, C# and Java.